

BAB IV HASIL DAN ANALISA

4.1 Pendahuluan

Bab ini membahas hasil implementasi dan pengujian sistem monitoring dan kontrol volume serta laju tetes infus berbasis *Internet of Things* yang telah dirancang. Pengujian dilakukan untuk mengetahui kinerja setiap komponen penyusun sistem, meliputi sensor *loadcell*, sensor *infrared*, motor servo, NodeMCU ESP32, serta komunikasi data menuju *platform* Openremote. Analisis dilakukan berdasarkan data hasil pengujian yang diperoleh selama penelitian. Data tersebut digunakan untuk mengevaluasi tingkat akurasi sensor, kemampuan sistem dalam melakukan monitoring secara *real-time*, kemampuan motor servo dalam mengontrol laju tetesan infus, serta waktu respons sistem dalam mengirimkan data ke aplikasi Openremote. Hasil pengujian kemudian dibandingkan dengan nilai acuan sehingga dapat diketahui tingkat keberhasilan sistem yang telah dirancang.

4.2 Pengujian Sensor *Loadcell*

4.4.1 Kalibrasi Sensor *Loadcell*

Kalibrasi bertujuan memperoleh *calibration factor* sehingga hasil pembacaan sensor mendekati massa sebenarnya. Nilai *calibration factor* diperoleh melalui beberapa kali percobaan hingga menghasilkan pembacaan yang stabil.

Tabel 4.1 Pengujian *Calibration Factor*

No	<i>Calibration factor</i>	<i>Loadcell</i> (gr)	Berat Referensi (gr)	Error (%)
1	-1000	514	544	5,51
2	-1030	524	544	3,68
3	-1050	536	544	1,47
4	-1060	542	544	0,37

4.4.2 Pengujian Ketepatan Pengukuran Massa

Pada tahap ini sensor *loadcell* berfungsi untuk mengukur atau mengetahui berapa volume dari isi cairan tabung infus. Sensor *loadcell* yang digunakan ukuran 1kg. Pengujian dilakukan dengan membandingkan hasil pembacaan sensor *loadcell* terhadap timbangan digital sebagai alat ukur acuan. Massa cairan infus diubah secara bertahap kemudian dicatat hasil pembacaan kedua alat. Selanjutnya dihitung nilai *error* menggunakan Persamaan berikut.

$$Error(\%) = \frac{|Sensor - Referensi|}{Referensi} \times 100\% \quad (4.1)$$



Gambar 4.1 Pengujian *loadcell* dengan timbangan

Tabel 4.2 Perbandingan data *loadcell* dan timbangan

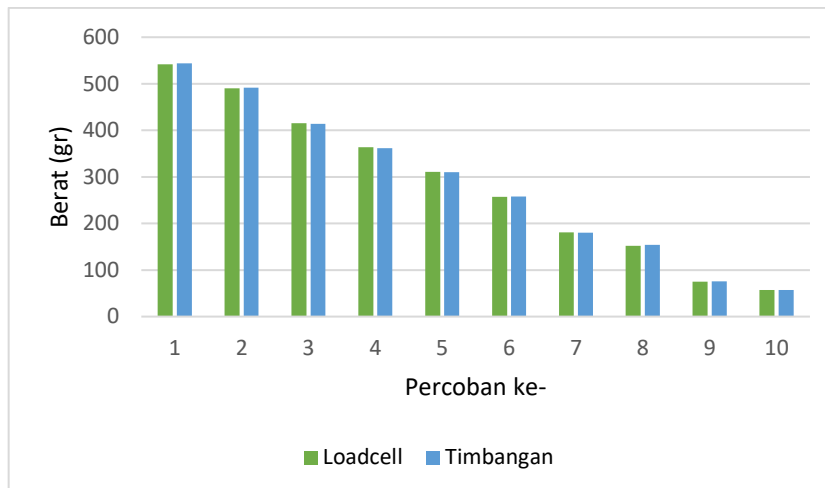
No	<i>Loadcell</i> (gr)	Timbangan (gr)	Selisih	Persentase <i>error</i> (%)
1	542	544	2	0,37
2	490	492	2	0,41
3	415	414	1	0,24
4	364	362	2	0,55
5	311	310	1	0,32
6	257	258	1	0,39
7	181	180	1	0,56
8	152	154	2	1,3
9	75	76	1	1,32
10	57	57	0	0

Berdasarkan data pada tabel 4.2, setelah didapatkan persentase *error* masing-masing percobaan, kemudian dicari *error* rata-rata menggunakan rumus berikut.

$$\text{rata - rata \%error} = \frac{\sum \%error}{\text{jumlah data}} \quad (4.2)$$

$$\text{rata - rata \%error} = \frac{5,45\%}{10} = 0,54\%$$

Persentase rata-rata *error* yang didapatkan adalah 0,724%. Oleh karena Persentase yang didapatkan kurang dari 5% maka pembacaan data berat infus oleh sensor *loadcell* cukup akurat.



Gambar 4.2 Grafik Perbandingan Nilai Massa

4.3 Pengujian Konversi Massa Menjadi Volume

Sensor *loadcell* menghasilkan keluaran berupa massa cairan infus. Agar dapat ditampilkan dalam satuan volume (mL), dilakukan proses konversi berdasarkan massa jenis cairan NaCl 0,9% yang mendekati 1 g/mL. Hasil konversi menunjukkan bahwa perubahan massa yang dibaca sensor memiliki hubungan linier terhadap perubahan volume cairan infus. Dengan demikian sistem mampu menampilkan sisa volume infus secara otomatis pada aplikasi Openremote.

4.4 Pengujian Sensor *Infrared*

Pengujian pengukuran laju tetes infus bertujuan untuk mengetahui ketepatan pembacaan laju tetes infus oleh sensor *infrared*. Pengujian dilakukan dengan membandingkan hasil pembacaan sensor *infrared* terhadap hasil perhitungan manual menggunakan *stopwatch* selama satu menit. Pengujian dilakukan pada beberapa variasi laju tetesan tanpa adanya interval tertentu.



Gambar 4.3 Pengujian Sensor *Infrared*

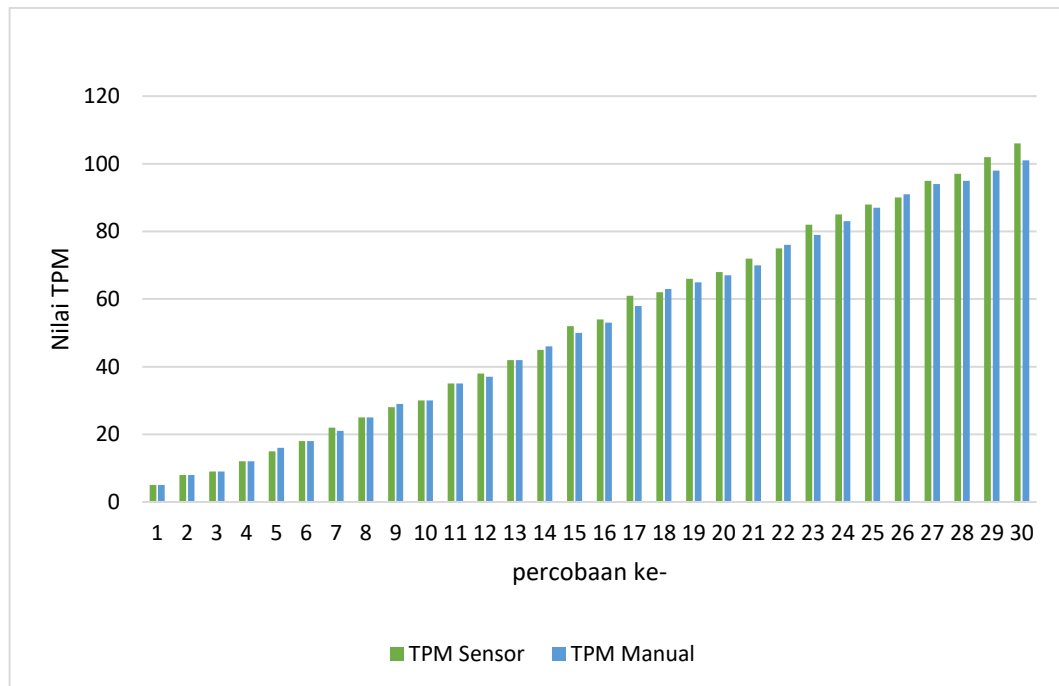
Tabel 4.3 Perbandingan TPM Sensor dan TPM Manual

No	TPM Infrared	TPM Manual	Selisih	Persentase error
1	5	5	0	0
2	8	8	0	0
3	9	9	0	0
4	12	12	0	0
5	15	16	1	6,25
6	18	18	0	0
7	22	21	1	4,76
8	25	25	0	0
9	28	29	1	3,45
10	30	30	0	0
11	35	35	0	0
12	38	37	1	2,70
13	42	42	0	0
14	45	46	1	2,17
15	52	50	2	4
16	54	53	1	1,89
17	61	58	3	5,17
18	62	63	1	1,59
19	66	65	1	1,54
20	68	67	1	1,49
21	72	70	2	2,86
22	75	76	1	1,32
23	82	79	3	3,80
24	85	83	2	2,41
25	88	87	1	1,15
26	90	91	1	1,10
27	95	94	1	1,06
28	97	95	2	2,11
29	102	98	4	4,08
30	106	101	5	4,95

Setelah didapatkan persentase *error* masing-masing data, kemudian dicari *error* rata-rata menggunakan persamaan 4.2.

$$\text{rata - rata \%error} = \frac{59,84\%}{30} = 1,99\%$$

Persentase rata-rata *error* yang didapatkan adalah 1,99%. Oleh karena persentase *error* yang didapatkan kurang dari 5% maka pembacaan laju tetesan infus oleh sensor *infrared* cukup akurat.



Gambar 4.4 Grafik Perbandingan Nilai TPM

4.5 Pengujian Motor Servo

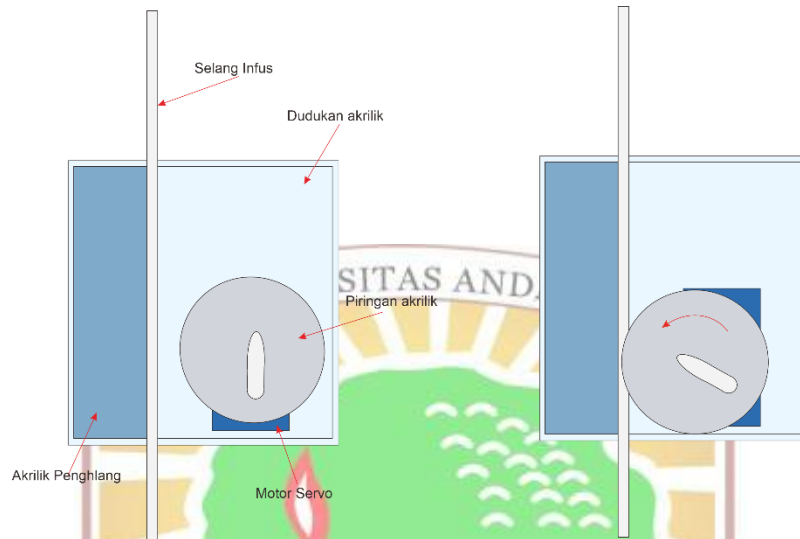
Motor servo diuji untuk mengetahui kemampuannya dalam mengatur bukaan klem infus sesuai nilai *setpoint* yang diberikan melalui aplikasi Openremote. Pengujian dilakukan pada beberapa *setpoint*, yaitu 10, 15, 20, 25, 30, 35, 40, 45, 50, dan 55` tetes per menit.



Gambar 4.5 Pengujian Motor Servo

Motor servo berfungsi sebagai aktuator yang mengatur bukaan pada selang infus melalui perubahan sudut putar yang dikendalikan oleh sinyal PWM dari ESP32. Perubahan sudut servo memengaruhi besar kecilnya bukaan selang sehingga menentukan laju aliran cairan dan jumlah tetesan per menit (TPM). Pada

sistem ini, sudut servo memiliki hubungan berbanding terbalik dengan nilai *setpoint* TPM, yaitu semakin besar sudut servo maka bukaan selang semakin kecil sehingga TPM menurun, sedangkan semakin kecil sudut servo maka bukaan selang semakin besar sehingga TPM meningkat. Oleh karena itu, pengendali PID mengubah sudut servo berdasarkan selisih antara nilai setpoint dan TPM aktual agar laju tetesan tetap sesuai dengan nilai yang diinginkan.



Gambar 4.6 Ilustrasi mekanisme pengontrolan laju tetes infus menggunakan motor servo

Tabel 4.4 Hasil pengujian motor servo

No	PWM (%)	Sudut Servo (°)	TPM <i>Setpoint</i>	TPM Sensor	Selisih	<i>Error</i> (%)
1	11,95	85	10	10	0	0,00
2	11,39	80	15	15	0	0,00
3	10,84	75	20	19	1	5,26
4	10,28	70	25	24	1	4,17
5	9,72	65	30	31	1	3,23
6	9,17	60	35	34	1	2,94
7	8,61	55	40	41	1	2,44
8	8,06	50	45	44	1	2,27
9	6,95	40	50	48	2	4,17
10	6,39	35	55	59	4	6,78

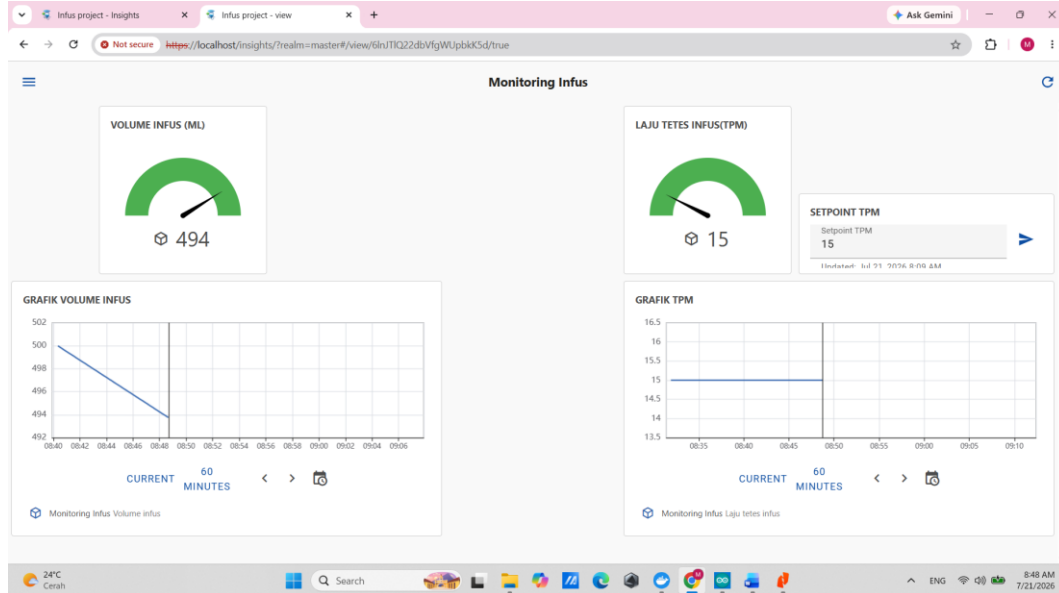
Setelah didapatkan persentase *error* masing-masing data, kemudian dicari *error* rata-rata menggunakan persamaan 4.2.

$$\text{rata - rata \%error} = \frac{31,25\%}{10} = 3,12\%$$

Persentase rata-rata *error* yang didapatkan adalah 3,12%. Oleh karena persentase *error* yang didapatkan kurang dari 5% maka motor servo cukup akurat.

4.6 Pengujian Monitoring pada Openremote

Pengujian ini bertujuan mengetahui kemampuan sistem dalam mengirimkan data sensor menuju *platform* Openremote menggunakan protokol MQTT. Parameter yang diamati meliputi keberhasilan koneksi, tampilan data volume, tampilan laju tetesan, serta perubahan status ketika volume infus mendekati habis.



Gambar 4.7 Pengujian Monitoring pada Openremote

4.7 Analisis Waktu Respons sistem

Pengujian waktu respons dilakukan untuk mengetahui selang waktu yang dibutuhkan sejak sensor membaca perubahan kondisi hingga data ditampilkan pada aplikasi Openremote. Waktu dihitung menggunakan *stopwatch* pada beberapa kali pengujian. Berdasarkan hasil pengujian, rata-rata waktu respons sistem berada pada rentang beberapa detik. Nilai tersebut dipengaruhi oleh kecepatan jaringan Wi-Fi, proses pengolahan data pada ESP32, komunikasi MQTT, dan proses pembaruan data pada *platform* Openremote. Semakin stabil koneksi internet yang digunakan maka waktu respons sistem semakin cepat.

Tabel 4.5 Waktu Respons Sistem

No	Percobaan	Waktu Respons (detik)
1	Pengujian 1	1,20
2	Pengujian 2	1,35
3	Pengujian 3	1,18
4	Pengujian 4	1,42
5	Pengujian 5	1,27
6	Pengujian 6	1,31
7	Pengujian 7	1,24
8	Pengujian 8	1,39
9	Pengujian 9	1,29

10	Pengujian 10	1,33
11	Pengujian 11	1,26
12	Pengujian 12	1,37
13	Pengujian 13	1,22
14	Pengujian 14	1,34
15	Pengujian 15	1,28
16	Pengujian 16	1,30
17	Pengujian 17	1,25
18	Pengujian 18	1,36
19	Pengujian 19	1,21
20	Pengujian 20	1,32

Berdasarkan hasil pengujian sebanyak 20 kali percobaan, diperoleh waktu respons sistem berkisar antara 1,18 detik hingga 1,42 detik dengan rata-rata sebesar 1,32 detik. Hasil tersebut menunjukkan bahwa sistem mampu mengirimkan data dari ESP32 ke *platform* Openremote melalui protokol MQTT dalam waktu yang relatif singkat. Perbedaan waktu respons pada setiap percobaan dipengaruhi oleh kondisi jaringan Wi-Fi, proses pengolahan data pada ESP32, komunikasi MQTT, serta proses pembaruan data pada *dashboard* Openremote. Dengan rata-rata waktu respons sebesar 1,32 detik, sistem telah mampu menampilkan data monitoring infus secara *real-time*, sehingga layak digunakan untuk mendukung pemantauan volume dan laju tetes infus.

4.8 Analisis Keseluruhan sistem

Secara keseluruhan sistem monitoring dan kontrol volume serta laju tetes infus telah mampu bekerja sesuai tujuan penelitian. Sensor *loadcell* mampu mengukur sisa volume cairan infus dengan tingkat kesalahan yang rendah, sedangkan sensor *infrared* mampu mendeteksi jumlah tetesan secara akurat. Motor servo juga dapat mengatur laju tetesan mendekati nilai *setpoint* yang diberikan. Integrasi NodeMCU ESP32 dengan *platform* Openremote melalui protokol MQTT memungkinkan proses monitoring dan kontrol dilakukan secara *real-time* melalui jaringan internet. Dengan demikian sistem yang dikembangkan dapat membantu tenaga medis dalam memantau kondisi infus tanpa harus melakukan pengecekan secara langsung secara terus-menerus, sehingga berpotensi meningkatkan efisiensi pelayanan dan keselamatan pasien.